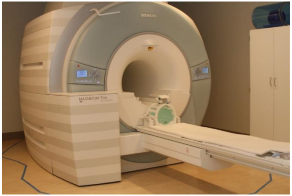
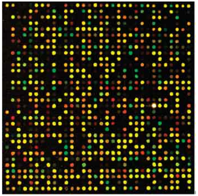
Compressed Subspace Matching, Blind Deconvolution, and Multichannel Sampling

Justin Romberg
Georgia Tech, ECE

CoSeRa 2013
September 19, 2013
Bonn, Germany

Linear systems of equations are ubiquitous





All of these can be abstracted to

$$\boldsymbol{A}\boldsymbol{x} = \boldsymbol{y}$$

What we know about solving linear systems

Observe:

$$\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x}_0 + \text{noise}$$

Classical: If $\boldsymbol{A}^{\mathrm{H}}\boldsymbol{A}$ is *well conditioned* then we can stably estimate $\boldsymbol{x}_0$ using *least-squares*.

Sparse: If $\boldsymbol{A}$ *keeps S-sparse signals separated* then we can stably estimate sparse $\boldsymbol{x}_0$ using *ℓ_1 minimization*.

Low rank: If $\boldsymbol{A}$ *keeps rank-R matrices separated* then we can stably estimate low-rank $\boldsymbol{x}_0$ using *nuclear norm minimization*.

The last two can be achieved for *underdetermined* $\boldsymbol{A}$ as long as its rows are *global and diverse*. This can be achieved by injecting *randomness* into $\boldsymbol{A}$.

Optimization programs for solving linear systems

Observe:

$$\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x}_0 + \text{noise}$$

Classical: *least-squares*:

$$\min_{\boldsymbol{x}} \ \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_2^2$$

Sparse: ℓ_1 *minimization*:

$$\min_{\boldsymbol{x}} \ \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_2^2 + \tau\|\boldsymbol{x}\|_1$$

where $\|\boldsymbol{x}\|_1 =$ sum of magnitudes

Low rank: *nuclear norm minimization*:

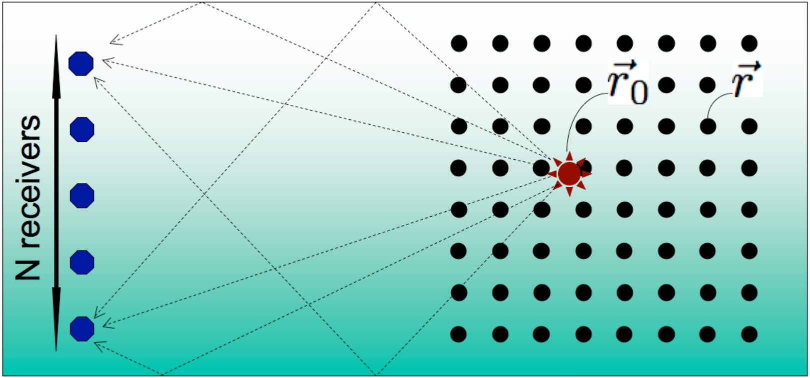
$$\min_{\boldsymbol{X}} \ \|\boldsymbol{y} - \mathcal{A}(\boldsymbol{X})\|_2^2 + \tau\|\boldsymbol{X}\|_*$$

where $\|\boldsymbol{X}\|_* =$ sum of singular values

Agenda for today

1. Compressive subspace matching on the continuum
2. Blind deconvolution using convex programming
3. Multichannel compressive sampling

Source localization

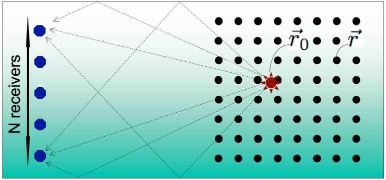


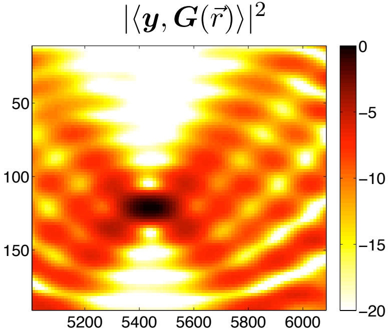
We observe a narrowband source emitting from (unknown) location $\vec{r_0}$:

$$\boldsymbol{y} = \alpha \boldsymbol{G}(\vec{r_0}) + \text{noise}, \quad \boldsymbol{y} \in \mathbb{C}^N$$

Goal: estimate $\vec{r_0}$ using only *implicit* knowledge of the channel $\boldsymbol{G}$

Matched field processing





Given observations $\boldsymbol{y}$, estimate $\vec{r_0}$ by "matching against the field":

$$\hat{r} = \arg\min_{\vec{r}} \min_{\beta \in \mathbb{C}} \|\boldsymbol{y} - \beta \boldsymbol{G}(\vec{r})\|^2 = \max_{\vec{r}} \frac{|\langle \boldsymbol{y}, \boldsymbol{G}(\vec{r})\rangle|^2}{\|\boldsymbol{G}(\vec{r})\|^2} \approx |\langle \boldsymbol{y}, \boldsymbol{G}(\vec{r})\rangle|^2$$

We do not have direct access to $\boldsymbol{G}$, but can calculate $\langle \boldsymbol{y}, \boldsymbol{G}(\vec{r})\rangle$ for all $\vec{r}$ using *time-reversal*

Coded simulations

- Pre-compute the responses to a series of *randomly and simultaneously activated* sources along the receiver array

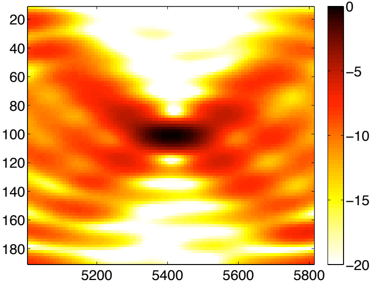
$$\boldsymbol{b}_1 = \boldsymbol{G}^{\mathrm{H}}\boldsymbol{\phi}_1, \quad \boldsymbol{b}_2 = \boldsymbol{G}^{\mathrm{H}}\boldsymbol{\phi}_2, \quad \ldots \quad \boldsymbol{b}_M = \boldsymbol{G}^{\mathrm{H}}\boldsymbol{\phi}_M,$$

  where the $\boldsymbol{\phi}_m$ are random vectors
- Stack up the $\boldsymbol{b}_m^{\mathrm{H}}$ to form the matrix $\boldsymbol{\Phi G}$
- Given the observations $\boldsymbol{y}$, code them to form $\boldsymbol{\Phi y}$, and solve

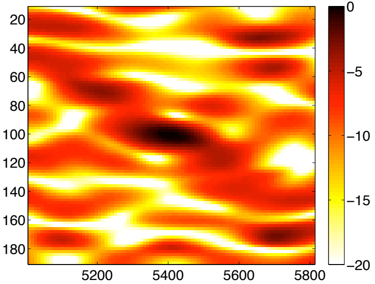
$$\hat{r}_{cs} = \arg\min_{\vec{r}} \min_{\beta\in\mathbb{C}} \|\boldsymbol{\Phi y} - \beta\boldsymbol{\Phi G}(\vec{r})\|_2^2 = \arg\max_{\vec{r}} \frac{|\langle \boldsymbol{\Phi y}, \boldsymbol{\Phi G}(\vec{r})\rangle|^2}{\|\boldsymbol{\Phi G}(\vec{r})\|^2}$$

Compressive ambiguity functions

ambiguity function $(\boldsymbol{G}^{\mathrm{H}}\boldsymbol{y})(\vec{r})$



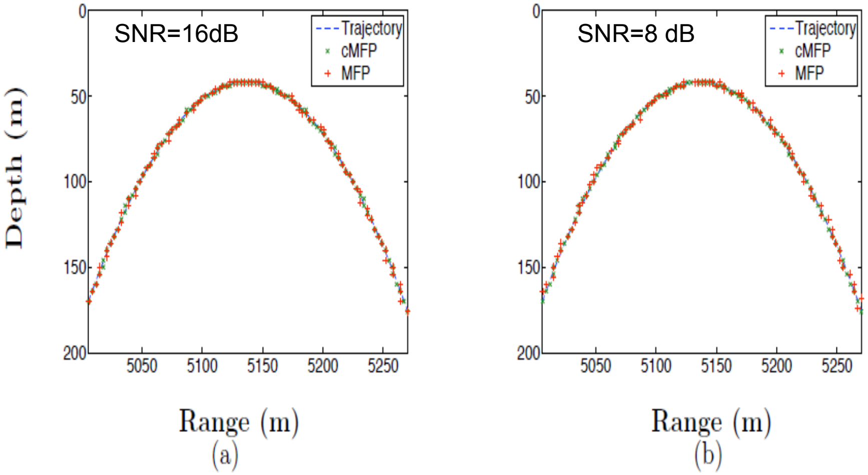
compressed amb func $(\boldsymbol{G}^{\mathrm{H}}\boldsymbol{\Phi}^{\mathrm{H}}\boldsymbol{\Phi}\boldsymbol{y})(\vec{r})$



$M = 10$ (compare to 37 receivers)

- The compressed ambiguity function is a *random process* whose mean is the true ambiguity function
- For very modest M, these two functions peak in the same place

Numerical simulation: source tracking



16x compression, very little loss in performance

Union of subspaces

Basic problem:
We have a collection of subspaces $\{\mathcal{S}_\theta,\ \theta \in \Theta\}$.
Given $\boldsymbol{y} = \boldsymbol{\Phi}\boldsymbol{x}_0$, we like to know which subspace is the best "fit" for $\boldsymbol{x}_0$

Applications:

1. source localization
2. direction of arrival estimation in array processing
3. pulse detection / time-of-arrival estimation from compressed samples ("smashed filtering")

Union of subspaces

Basic problem:
We have a collection of subspaces $\{\mathcal{S}_\theta,\ \theta \in \Theta\}$ in $\mathbb{R}^N$.
Given $\boldsymbol{y} = \boldsymbol{\Phi}\boldsymbol{x}_0$, we like to know which subspace is the best "fit" for $\boldsymbol{x}_0$

Two questions:

1. When can we distinguish $\boldsymbol{x}_1 \in \mathcal{S}_{\theta_1}$ and $\boldsymbol{x}_2 \in \mathcal{S}_{\theta_2}$ when viewed through $\boldsymbol{\Phi}$? *Stable embedding:*

$$(1-\delta)\|\boldsymbol{x}_1 - \boldsymbol{x}_2\|_2^2 \ \leq\ \|\boldsymbol{\Phi}\boldsymbol{x}_1 - \boldsymbol{\Phi}\boldsymbol{x}_2\|_2^2 \ \leq\ (1+\delta)\|\boldsymbol{x}_1 - \boldsymbol{x}_2\|_2^2$$

2. When can we find subspace most closely aligned with $\boldsymbol{x}_0$ when viewed through $\boldsymbol{\Phi}$?

Embedding subsets of $\mathbb{R}^N$

Let $\mathcal{Q} \subset \mathbb{R}^N$. For $\mathbf{\Phi}$ random, when do we have

$$(1-\delta)\|\boldsymbol{x}_1 - \boldsymbol{x}_2\|_2^2 \;\le\; \|\mathbf{\Phi}\boldsymbol{x}_1 - \mathbf{\Phi}\boldsymbol{x}_2\|_2^2 \;\le\; (1+\delta)\|\boldsymbol{x}_1 - \boldsymbol{x}_2\|_2^2,$$

for all $\boldsymbol{x}_1, \boldsymbol{x}_2 \in \mathcal{Q}$ with appropriately high probability?

- $\mathcal{Q}$ is a finite set of size $|\mathcal{Q}| = Q$. Then

  $$\delta \;\lesssim\; \sqrt{\frac{2\log Q}{M}}.$$

  So we can take

  $$M \gtrsim 2\log Q$$

  This is known as the *Johnson-Lindenstrauss Lemma* (1984).

Embedding subsets of $\mathbb{R}^N$

Let $\mathcal{Q} \subset \mathbb{R}^N$. For $\mathbf{\Phi}$ random, when do we have

$$(1-\delta)\|\boldsymbol{x}_1 - \boldsymbol{x}_2\|_2^2 \;\le\; \|\mathbf{\Phi}\boldsymbol{x}_1 - \mathbf{\Phi}\boldsymbol{x}_2\|_2^2 \;\le\; (1+\delta)\|\boldsymbol{x}_1 - \boldsymbol{x}_2\|_2^2,$$

for all $\boldsymbol{x}_1, \boldsymbol{x}_2 \in \mathcal{Q}$ with appropriately high probability?

- $\mathcal{Q}$ is a subspace of dimension K. Then δ is directly related to the *singular values* of $\mathbf{\Phi}$, and

  $$\delta \;\lesssim\; \sqrt{\frac{K}{M}},$$

  so we can take

  $$M \;\gtrsim\; K$$

  This is a "classical" result by Marchenko, Pastur (1960s), and later Szarek (1990s).

Embedding subsets of $\mathbb{R}^N$

Let $\mathcal{Q} \subset \mathbb{R}^N$. For $\mathbf{\Phi}$ random, when do we have

$$(1-\delta)\|\boldsymbol{x}_1 - \boldsymbol{x}_2\|_2^2 \ \leq \ \|\mathbf{\Phi}\boldsymbol{x}_1 - \mathbf{\Phi}\boldsymbol{x}_2\|_2^2 \ \leq \ (1+\delta)\|\boldsymbol{x}_1 - \boldsymbol{x}_2\|_2^2,$$

for all $\boldsymbol{x}_1, \boldsymbol{x}_2 \in \mathcal{Q}$ with appropriately high probability?

- $\mathcal{Q}$ is a finite collection of subspaces of dimension K, $\{\mathcal{S}_\theta, \ \theta \in \Theta$. Then

  $$\delta \ \lesssim \ \sqrt{\frac{2K + 2\log|\Theta|}{M}}$$

  Example: sparse recovery for compressive sensing, $|\Theta| = \binom{N}{K} \sim e^{K\log(N/K)}$, and so we can take

  $$M \ \gtrsim \ 2K\log(N/K)$$

  (Candes,Tao; Rudleson, Vershynin; Davenport et al., mid-2000s)

Embedding subsets of $\mathbb{R}^N$

Let $\mathcal{Q} \subset \mathbb{R}^N$. For $\mathbf{\Phi}$ random, when do we have

$$(1-\delta)\|\boldsymbol{x}_1 - \boldsymbol{x}_2\|_2^2 \;\leq\; \|\mathbf{\Phi}\boldsymbol{x}_1 - \mathbf{\Phi}\boldsymbol{x}_2\|_2^2 \;\leq\; (1+\delta)\|\boldsymbol{x}_1 - \boldsymbol{x}_2\|_2^2,$$

for all $\boldsymbol{x}_1, \boldsymbol{x}_2 \in \mathcal{Q}$ with appropriately high probability?

- $\mathcal{Q}$ is a smooth manifold of dimension K. Then

$$\delta \;\lesssim\; \sqrt{\frac{2K \cdot f(\text{curvature, volume,etc.})}{M}}$$

(Wakin et al, Woodruff, Yap et al, ..., recent)

Embedding subsets of $\mathbb{R}^N$

Let $\mathcal{Q} \subset \mathbb{R}^N$. For $\mathbf{\Phi}$ random, when do we have

$$(1-\delta)\|\boldsymbol{x}_1 - \boldsymbol{x}_2\|_2^2 \;\le\; \|\mathbf{\Phi}\boldsymbol{x}_1 - \mathbf{\Phi}\boldsymbol{x}_2\|_2^2 \;\le\; (1+\delta)\|\boldsymbol{x}_1 - \boldsymbol{x}_2\|_2^2,$$

for all $\boldsymbol{x}_1, \boldsymbol{x}_2 \in \mathcal{Q}$ with appropriately high probability?

- $\mathcal{Q}$ is a infinite collection of subspaces of dimension K, $\{\mathcal{S}_\theta : \theta \in \Theta\}$. We can take

  $$\delta \;\lesssim\; \sqrt{\frac{2K\Delta}{M}}$$

  where Δ is a measure of *geometrical complexity* of Θ.

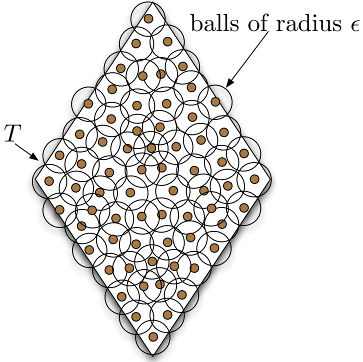
  In typical cases of interest, $\Delta \sim \log\left(\max(K, \text{“effective dimension”})\right)$.

  (Mantzel and R. '13)

Geometrical complexity

Covering numbers:

$N(T, d, \epsilon)$ $=$ size of smallest ϵ-cover



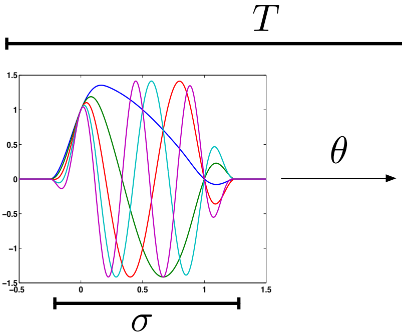
We have $T = \{\mathcal{S}_\theta\}_\theta$, $\quad d(\mathcal{S}_{\theta_1}, \mathcal{S}_{\theta_2}) = \|\boldsymbol{P}_{\theta_1} - \boldsymbol{P}_{\theta_2}\|$

Then Δ depends on how fast the cover grows as $\epsilon \to 0$, characterized by N_0, d such that

$$N(T, d, \epsilon) \;\leq\; N_0 \left(\frac{1}{\epsilon}\right)^d$$

Example: Shiftable subspaces

Smooth window, modulated by K different cosines (LOT).
Width of functions $= \sigma$
Shift over interval of length T



In this case, we have

$$\Delta \sim \log(K) + \log(T/\sigma)$$

Compressive subspace matching

Collection of subspaces $\{\mathcal{S}_\theta, \theta \in \Theta\}$
Observe $\boldsymbol{y} = \boldsymbol{\Phi x}$, where $\boldsymbol{x} \in \mathcal{S}_{\theta_0}$

Full observation: Solve

$$\bar{\theta} = \arg\min_{\theta \in \Theta} \; \|\boldsymbol{x} - \boldsymbol{P}_\theta \boldsymbol{x}\|_2^2$$

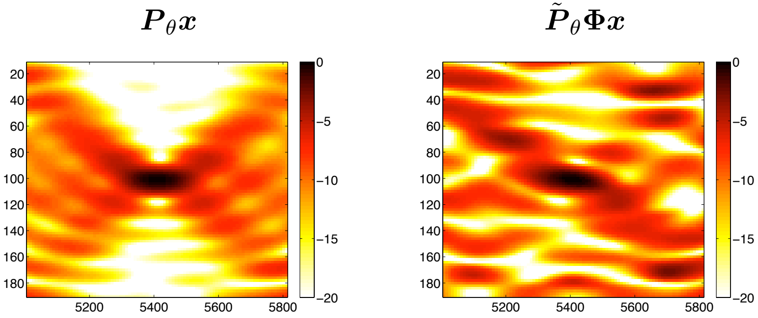
where $\boldsymbol{P}_\theta = \boldsymbol{V}_\theta \boldsymbol{V}_\theta^{\mathrm{T}}$ is the projector onto $\mathcal{S}_\theta$.

Compressed observation: Solve

$$\hat{\theta} = \arg\min_{\theta \in \Theta} \; \|\boldsymbol{y} - \tilde{\boldsymbol{P}}_\theta \boldsymbol{y}\|_2^2$$

where $\tilde{\boldsymbol{P}}_\theta = \boldsymbol{\Phi} \boldsymbol{V}_\theta (\boldsymbol{V}_\theta^{\mathrm{T}} \boldsymbol{\Phi}^{\mathrm{T}} \boldsymbol{\Phi} \boldsymbol{V}_\theta)^{-1} \boldsymbol{V}_\theta^{\mathrm{T}} \boldsymbol{\Phi}^{\mathrm{T}}$

Compressive subspace matching



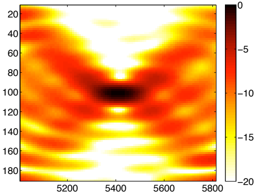
$$\bar{\theta} = \arg\max_\theta \|\boldsymbol{P}_\theta \boldsymbol{x}\|_2^2 \qquad \hat{\theta} = \arg\max_\theta \|\tilde{\boldsymbol{P}}_\theta \boldsymbol{y}\|_2^2$$

Performance gap:

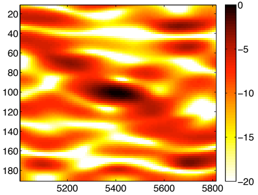
$$\hat{E} - \bar{E} = \|\boldsymbol{P}_{\bar{\theta}} \boldsymbol{x}\|_2^2 - \|\boldsymbol{P}_{\hat{\theta}} \boldsymbol{x}\|_2^2$$

Compressive subspace matching

$\boldsymbol{P}_\theta \boldsymbol{x}$



$\tilde{\boldsymbol{P}}_\theta \boldsymbol{\Phi} \boldsymbol{x}$



Performance gap (for $\|\boldsymbol{x}\|_2 = 1$):

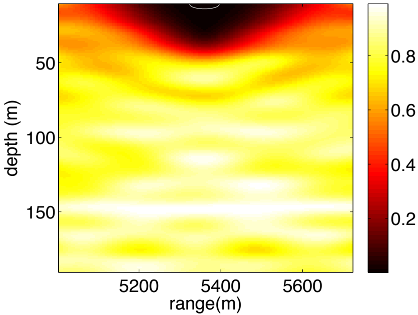
$$\hat{E} - \bar{E} \le \sqrt{\frac{K\Delta}{M}}$$

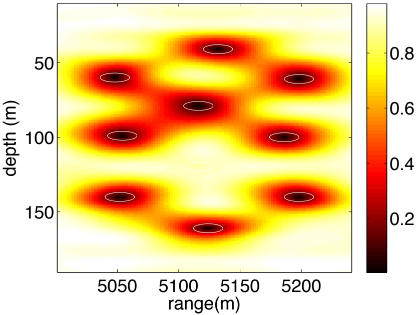
where Δ is the same geometric constant as before.

Compressive subspace matching is effective for

$$M \gtrsim K \log\left(\max(K, \text{“fill factor”})\right)$$

Underwater acoustics: multiple sources

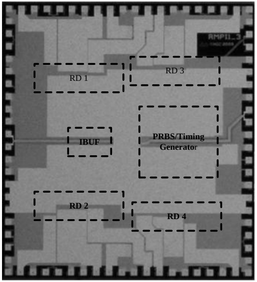


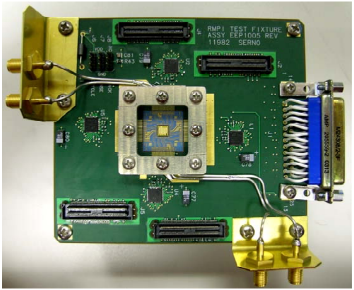


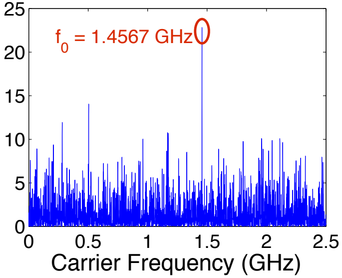
Right: strong source at surface washes out 9 weaker sources

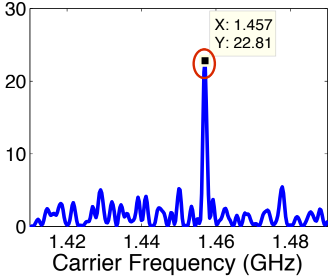
Left: locating then nulling out strong source flushed out weaker ones

Frequency estimation on actual hardware

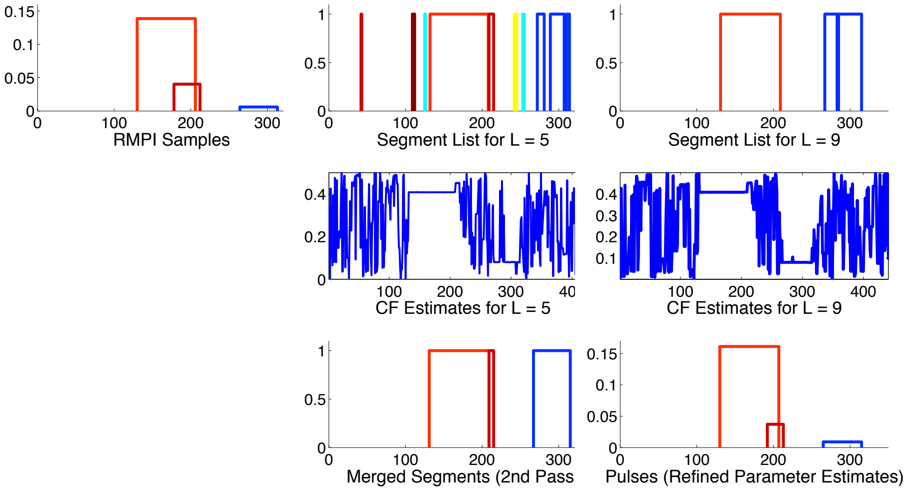






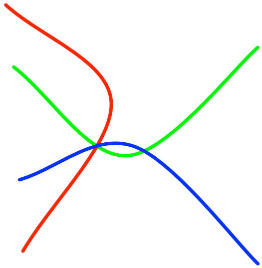


Pulse detection and segmentation



Blind deconvolution using convex programming

Bilinear equations



Bilinear equations contain unknown terms multiplied by one another

$$u_1v_1 + 5u_1v_2 + 7u_2v_3 = -12$$

$$u_3v_1 - 9u_2v_2 + 4u_3v_2 = 2$$

$$u_1v_2 - 6u_1v_3 - u_3v_3 = 7$$

Their nonlinearity makes them trickier to solve, and the computational framework is nowhere nearly as strong as for linear equations

Bilinear equations

Simple (but only recently appreciated) observation:
Systems of bilinear equations, e. g.

$$u_1 v_1 + 5 u_1 v_2 + 7 u_2 v_3 = -12$$
$$u_3 v_1 - 9 u_2 v_2 + 4 u_3 v_2 = 2$$

can be recast as *linear system of equations on a matrix that has rank 1*:

$$uv^T = \begin{bmatrix} u_1 v_1 & u_1 v_2 & u_1 v_3 & \cdots & u_1 v_N \\ u_2 v_1 & u_2 v_2 & u_2 v_3 & \cdots & u_2 v_N \\ u_3 v_1 & u_3 v_2 & u_3 v_3 & \cdots & u_3 v_N \\ \vdots & \vdots & & \ddots & \\ u_K v_1 & u_K v_2 & u_K v_3 & \cdots & u_K v_N \end{bmatrix}$$

Bilinear equations

Simple (but only recently appreciated) observation:
Systems of bilinear equations, e. g.

$$u_1 v_1 + 5u_1 v_2 + 7u_2 v_3 = -12$$
$$u_3 v_1 - 9u_2 v_2 + 4u_3 v_2 = 2$$

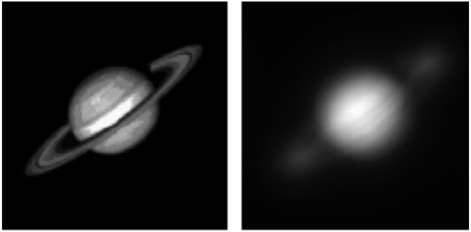
can be recast as *linear system of equations on a matrix that has rank 1*:

$$uv^T = \begin{bmatrix} u_1v_1 & u_1v_2 & u_1v_3 & \cdots & u_1v_N \\ u_2v_1 & u_2v_2 & u_2v_3 & \cdots & u_2v_N \\ u_3v_1 & u_3v_2 & u_3v_3 & \cdots & u_3v_N \\ \vdots & \vdots & & \ddots & \\ u_Kv_1 & u_Kv_2 & u_Kv_3 & \cdots & u_Kv_N \end{bmatrix}$$

Compressive (low rank) recovery $\Rightarrow$
"Generic" quadratic systems with cN equations and N unknowns can be solved using nuclear norm minimization

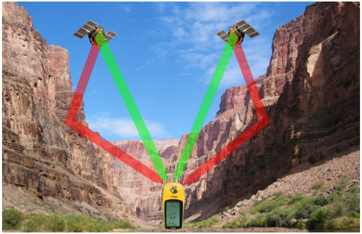
Blind deconvolution

image deblurring



(image courtesy of Hao, Lu, Qinzhang)

multipath in wireless comm



(image from Wikimedia Commons)

We observe

$$y[\ell] = \sum_n s[n]\, h[\ell - n]$$

and want to "untangle" $\boldsymbol{s}$ and $\boldsymbol{h}$.

Blind deconvolution as low rank recovery

Each sample of $\boldsymbol{y} = \boldsymbol{s} * \boldsymbol{h}$ is a bilinear combination of the unknowns,

$$y[\ell] = \sum_n s[n]h[\ell - n]$$

and is a *linear* combination of $\boldsymbol{s}\boldsymbol{h}^{\mathrm{T}}$:

$$\begin{array}{r c}
 & \\
 & \\
y_1[0] & \\
y_1[1] & \\
y_1[2] & \\
 & \\
\vdots & \\
 & \\
 & \\
 & \\
 & \\
y_1[9] &
\end{array}
\begin{bmatrix}
s[-2]h[0] & s[-2]h[1] & s[-2]h[2] \\
s[-1]h[0] & s[-1]h[1] & s[-1]h[2] \\
s[0]h[0] & s[0]h[1] & s[0]h[2] \\
s[1]h[0] & s[1]h[1] & s[1]h[2] \\
s[2]h[0] & s[2]h[1] & s[2]h[2] \\
s[3]h[0] & s[3]h[1] & s[3]h[2] \\
s[4]h[0] & s[4]h[1] & s[4]h[2] \\
s[5]h[0] & s[5]h[1] & s[5]h[2] \\
s[6]h[0] & s[6]h[1] & s[6]h[2] \\
s[7]h[0] & s[7]h[1] & s[7]h[2] \\
s[8]h[0] & s[8]h[1] & s[8]h[2] \\
s[9]h[0] & s[9]h[1] & s[9]h[2]
\end{bmatrix}$$

Blind deconvolution as low rank recovery

Given $\boldsymbol{y} = \boldsymbol{s} * \boldsymbol{h}$, it is impossible to untangle $\boldsymbol{s}$ and $\boldsymbol{h}$ unless we make some *structural assumptions*

Structure: $\boldsymbol{s}$ and $\boldsymbol{h}$ live in known *subspaces* of $\mathbb{R}^L$; we can write

$$\boldsymbol{s} = \boldsymbol{B}\boldsymbol{u}, \qquad \boldsymbol{h} = \boldsymbol{C}\boldsymbol{v}, \quad B : L \times K, \;\; C : L \times N$$

where $\boldsymbol{B}$ and $\boldsymbol{C}$ are matrices whose columns form bases for these spaces

We can now write blind deconvolution as a *linear inverse problem with a rank contraint*:

$$\boldsymbol{y} = \mathcal{A}(\boldsymbol{X}_0), \quad \text{where} \;\; \boldsymbol{X}_0 = \boldsymbol{s}\boldsymbol{h}^{\mathrm{T}} \;\; \text{has rank=1}$$

The action of $\mathcal{A}(\cdot)$ can be broken down into three linear steps:

$$\boldsymbol{X}_0 \;\rightarrow\; \boldsymbol{B}\boldsymbol{X}_0 \;\rightarrow\; \boldsymbol{B}\boldsymbol{X}_0\boldsymbol{C}^{\mathrm{T}} \;\rightarrow\; \text{take skew-diagonal sums}$$

Blind deconvolution theoretical results

We observe

$$\begin{aligned} \boldsymbol{y} &= \boldsymbol{s} * \boldsymbol{h}, \qquad \boldsymbol{h} = \boldsymbol{B}\boldsymbol{w}, \quad \boldsymbol{s} = \boldsymbol{C}\boldsymbol{x} \\ &= \mathcal{A}(\boldsymbol{w}\boldsymbol{x}^{\mathrm{T}}), \qquad \boldsymbol{w} \in \mathbb{R}^K, \quad \boldsymbol{x} \in \mathbb{R}^N, \end{aligned}$$

and then solve

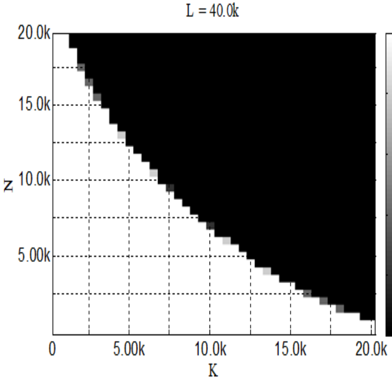
$$\min_{\boldsymbol{X}} \; \|\boldsymbol{X}\|_* \quad \text{subject to} \quad \mathcal{A}(\boldsymbol{X}) = \boldsymbol{y}.$$

Ahmed, Recht, R, '12:
If $\boldsymbol{B}$ is "incoherent" in the Fourier domain, and $\boldsymbol{C}$ is randomly chosen, then we will recover $\boldsymbol{X}_0 = \boldsymbol{s}\boldsymbol{h}^{\mathrm{T}}$ exactly (with high probability) when

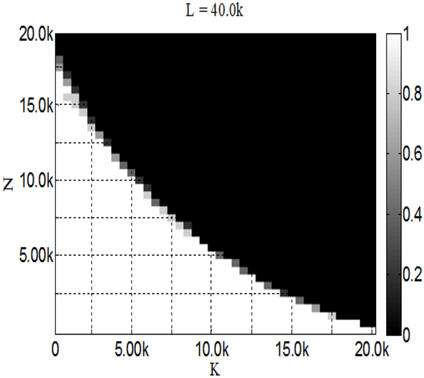
$$L \;\geq\; \mathrm{Const} \cdot (K + N) \log^3(KN)$$

Numerical results

white $=$ 100% success, black $=$ 0% success



$\boldsymbol{h}$ sparse, $\boldsymbol{s}$ sparse



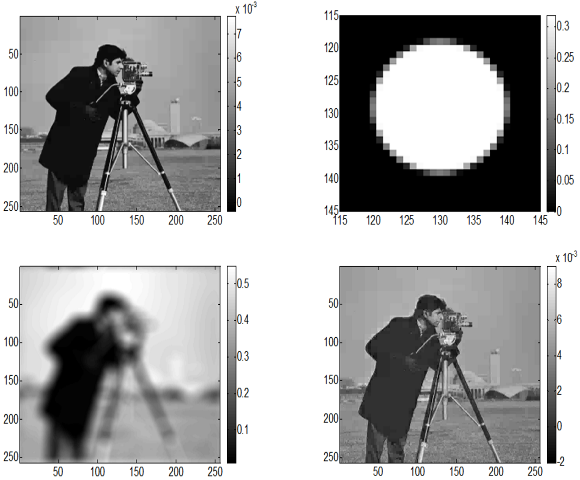
$\boldsymbol{h}$ sparse, $\boldsymbol{s}$ short

In the cases above, we can take

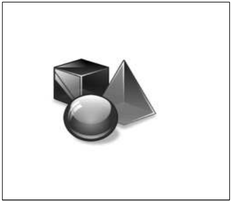
$$(N + K) \lesssim L/3$$

Numerical results

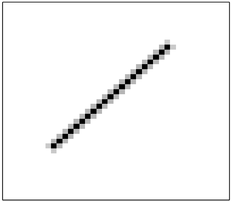
Unknown image with known support in the wavelet domain,
Unknown blurring kernel with known support in spatial domain



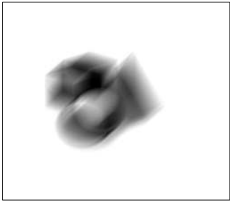
Numerical results



image



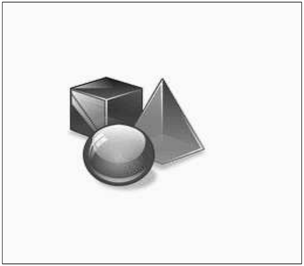
blurring kernel



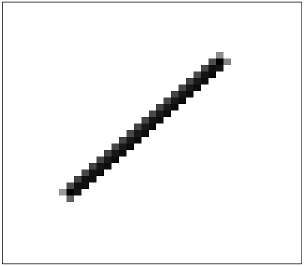
blurred image

Numerical results

Oracle recovery



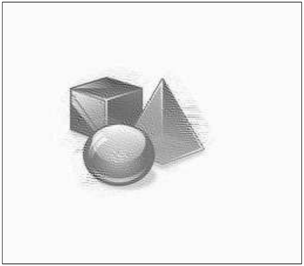
recovered image



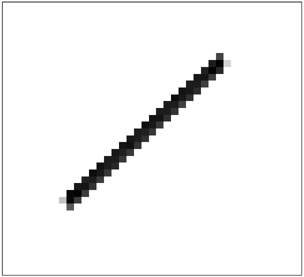
recovered kernel

Numerical results

Adaptive recovery

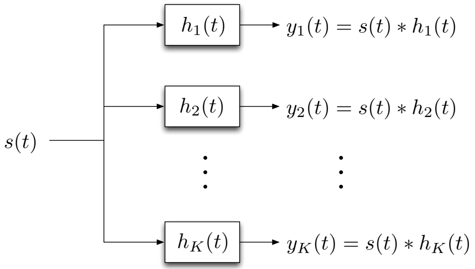


recovered image



recovered kernel

Passive imaging with multiple channels

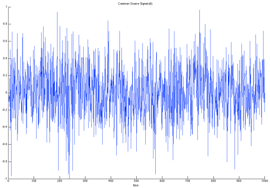


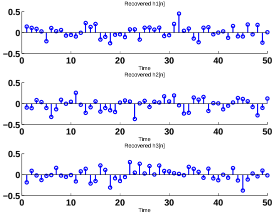
$$
\begin{array}{c}
\\ \\ y_1[0] \\ y_1[1] \\ y_1[2] \\ \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ y_1[9]
\end{array}
\left[
\begin{array}{ccc|ccc|ccc}
s[-2]h_1[0] & s[-2]h_1[1] & s[-2]h_1[2] & s[-2]h_2[0] & s[-2]h_2[1] & s[-2]h_2[2] & s[-2]h_3[0] & s[-2]h_3[1] & s[-2]h_3[2] \\
s[-1]h_1[0] & s[-1]h_1[1] & s[-1]h_1[2] & s[-1]h_2[0] & s[-1]h_2[1] & s[-1]h_2[2] & s[-1]h_3[0] & s[-1]h_3[1] & s[-1]h_3[2] \\
s[0]h_1[0] & s[0]h_1[1] & s[0]h_1[2] & y_2[0]\; s[0]h_2[0] & s[0]h_2[1] & s[0]h_2[2] & y_3[0]\; s[0]h_3[0] & s[0]h_3[1] & s[0]h_3[2] \\
s[1]h_1[0] & s[1]h_1[1] & s[1]h_1[2] & y_2[1]\; s[1]h_2[0] & s[1]h_2[1] & s[1]h_2[2] & y_3[1]\; s[1]h_3[0] & s[1]h_3[1] & s[1]h_3[2] \\
s[2]h_1[0] & s[2]h_1[1] & s[2]h_1[2] & y_2[2]\; s[2]h_2[0] & s[2]h_2[1] & s[2]h_2[2] & y_3[2]\; s[2]h_3[0] & s[2]h_3[1] & s[2]h_3[2] \\
s[3]h_1[0] & s[3]h_1[1] & s[3]h_1[2] & \cdot\; s[3]h_2[0] & s[3]h_2[1] & s[3]h_2[2] & \cdot\; s[3]h_3[0] & s[3]h_3[1] & s[3]h_3[2] \\
s[4]h_1[0] & s[4]h_1[1] & s[4]h_1[2] & \cdot\; s[4]h_2[0] & s[4]h_2[1] & s[4]h_2[2] & \cdot\; s[4]h_3[0] & s[4]h_3[1] & s[4]h_3[2] \\
s[5]h_1[0] & s[5]h_1[1] & s[5]h_1[2] & \cdot\; s[5]h_2[0] & s[5]h_2[1] & s[5]h_2[2] & \cdot\; s[5]h_3[0] & s[5]h_3[1] & s[5]h_3[2] \\
s[6]h_1[0] & s[6]h_1[1] & s[6]h_1[2] & \cdot\; s[6]h_2[0] & s[6]h_2[1] & s[6]h_2[2] & \cdot\; s[6]h_3[0] & s[6]h_3[1] & s[6]h_3[2] \\
s[7]h_1[0] & s[7]h_1[1] & s[7]h_1[2] & \cdot\; s[7]h_2[0] & s[7]h_2[1] & s[7]h_2[2] & \cdot\; s[7]h_3[0] & s[7]h_3[1] & s[7]h_3[2] \\
s[8]h_1[0] & s[8]h_1[1] & s[8]h_1[2] & s[8]h_2[0] & s[8]h_2[1] & s[8]h_2[2] & s[8]h_3[0] & s[8]h_3[1] & s[8]h_3[2] \\
s[9]h_1[0] & s[9]h_1[1] & s[9]h_1[2] & y_2[9]\; s[9]h_2[0] & s[9]h_2[1] & s[9]h_2[2] & y_3[9]\; s[9]h_3[0] & s[9]h_3[1] & s[9]h_3[2]
\end{array}
\right]
$$

Recovery results

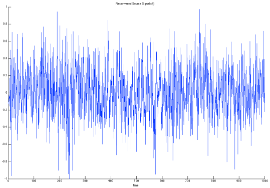
Source / output length: 1000
Number of channels: 100
Channel impulse response length: 50

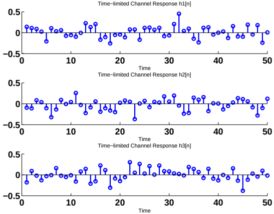
Original:





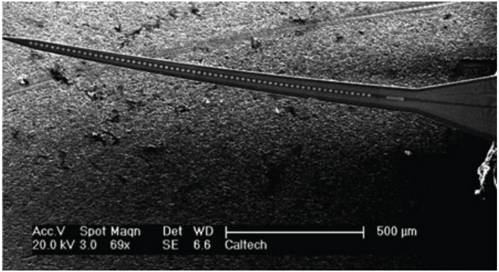
Recovered:



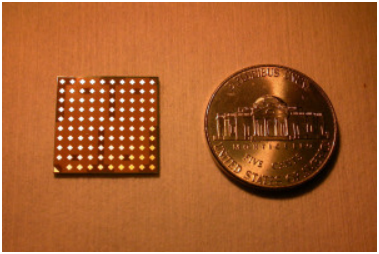


Sampling correlated signals

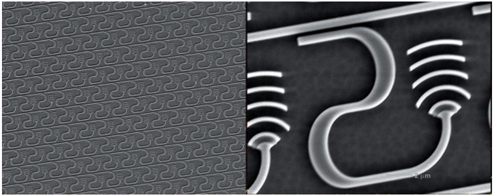
Sensor arrays



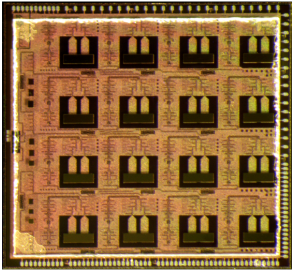
Caltech multielectrode



IBM phased array

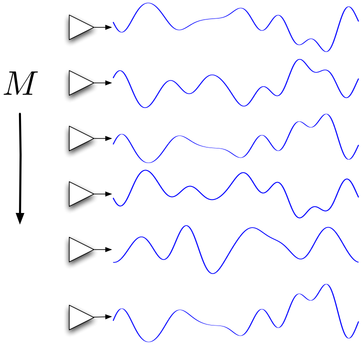


MIT nanophotonic array



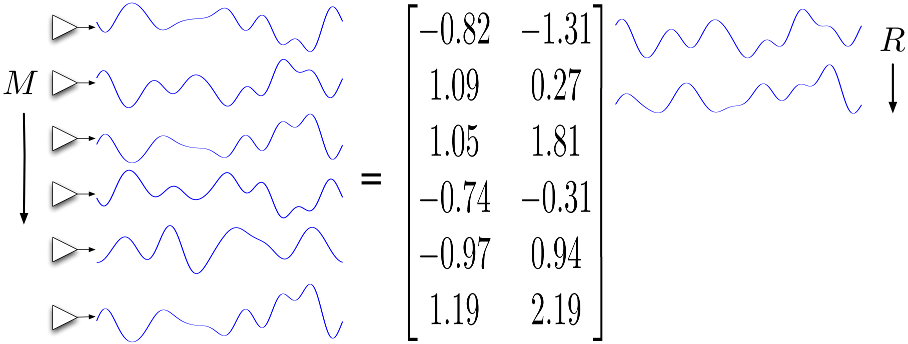
UCSD phased

Sampling correlated signals



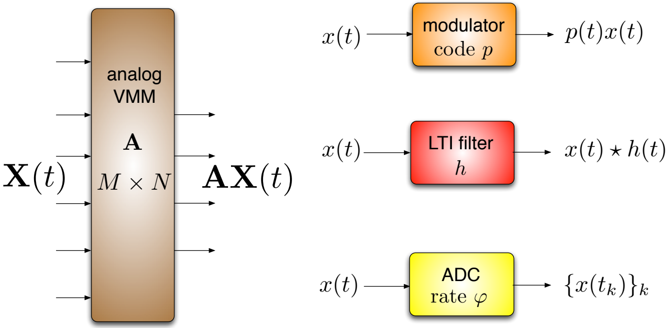
- Goal: acquire an *ensemble* of M signals
- Bandlimited to $W/2$
- "Correlated" $\rightarrow$ M signals are $\approx$ linear combinations of R signals

Sampling correlated signals



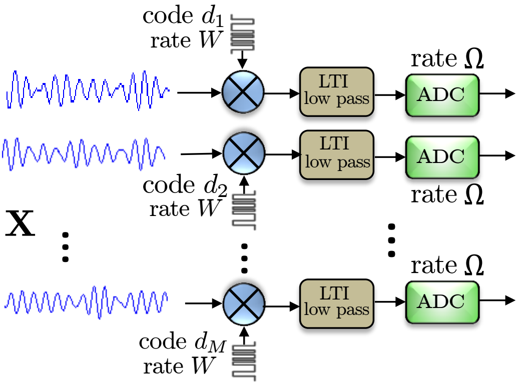
- Goal: acquire an *ensemble* of M signals
- Bandlimited to $W/2$
- "Correlated" $\to$ M signals are $\approx$ linear combinations of R signals

Components



- Analog vector-matrix multiplier spreads energy across channels
- Modulators spread energy across frequency
- Filters spread energy in one channel across time
- ADCs take samples

Sampling using the random demodulator



- Instead of running each ADC at rate $\Omega \geq W$, we can take

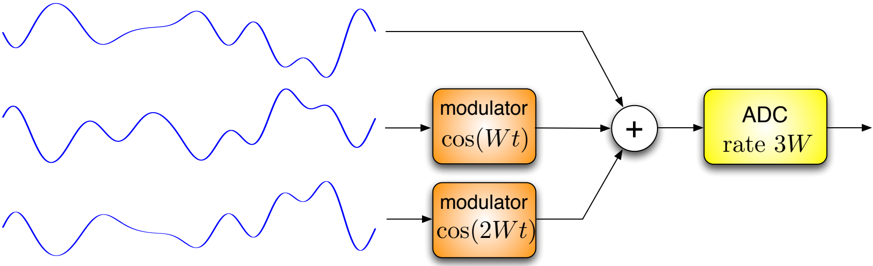
$$\Omega \gtrsim \frac{R}{M} W$$

to within logarithmic factors

Multiplexing onto one channel

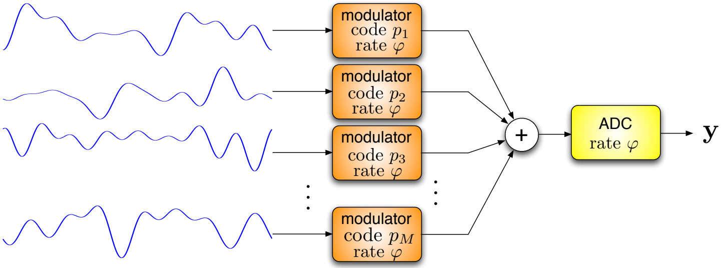
- We can always combine M channels into 1 by *multiplexing* in either time or frequency

  Frequency multiplexer:



- Replace M ADCs running at rate W with 1 ADC at rate MW

Compressive multiplexing



- If the signals are somewhat spread out in time, then the ADC and modulators can run at rate

$$\varphi \gtrsim RW$$

to within logarithmic factors

References

- W. Mantzel, K. Sabra, J. Romberg, "Compressive matched-field processing," *J. Acoustic Soc. Amer.*, July 2012.
- W. Mantzel and J. Romberg, "Compressed subspace matching on the continuum," manuscript to be submitted to *IEEE Trans. PAMI*, 2013.
- A. Ahmed and J. Romberg, "Compressive Sampling of Correlated Signals," Manuscript in preparation.
  Preliminary version at *IEEE Asilomar Conference on Signals, Systems, and Computers*, October 2011.
- A. Ahmed and J. Romberg, "Compressive Multiplexers for Correlated Signals," submitted to *IEEE Transactions on Information Theory*, August 2013.
- A. Ahmed, B. Recht, and J. Romberg, "Blind Deconvolution using Convex Programming," to appear in *IEEE Transactions on Information Theory*, late 2013.

`http://users.ece.gatech.edu/~justin/Publications.html`